

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions of claims in the application:

Listing of Claims:

1. (Currently Amended) A ~~Web-based system that asynchronously processes synchronous requests, comprising:~~
a processor that executes the following computer executable components stored on a computer readable medium:
an interface component that ~~synchronously receives a web-based synchronous request and synchronously returns a result based on the web-based request; [[and]]~~
a query management component that determines available processing engine computing capacity and predicts future processing engine capacity; and
a processing component that allocates parts of~~parses the web-based synchronous request across~~among processing engines based on the determined and predicted processing engine capacity and derives the result by aggregating processing engine results~~a plurality of Web services for asynchronous processing, wherein the processing component aggregates asynchronous results from the plurality of Web services and returns a synchronous result.~~
2. (Currently Amended) The system of claim 1, wherein the processing component automatically reallocates one or more parts of the web-based request serviced by a failed processing engine to one or more other processing engines~~parses the synchronous request based on a load balancing technique that distributes portions of the synchronous request to one or more of the plurality of Web services so that request processing is spread across respective Web services based on Web service load.~~

3. (Currently Amended) The system of claim 2, wherein the processing component~~load balancing technique~~ dynamically allocates one or more parts of the web-based request from a first processing engine to a second processing engine, wherein the processing engine capacity of the second processing engine is greater than the processing engine capacity of the first processing engine~~conveys portions of the request from a first Web service to a second Web service with a lesser load, during processing.~~

4. (Canceled)

5. (Currently Amended) The system of claim 1, wherein the web-based~~synchronous~~ request and result are~~are~~ conveyed by~~across~~ the interface component *via* at least one of the following protocols: TCP/IP; IPX/SPX; UDP/IP; HTTP; SOAP; or a proprietary protocol.

6. (Currently Amended) The system of claim 1, wherein the query management component stores the web-based request in a queue, and wherein the web-based request is serviced by one or more processing engines~~further comprises a queue that is utilized to post the synchronous request for retrieval by one or more Web services that are subscribed to process web-based requests stored in the queue.~~

7. (Currently Amended) The system of claim 6, wherein the queue is utilized to store information indicative of ~~at least one of a querying client, the synchronous request, the interface component, the processing component, the processing component queue, or a connection type,~~ wherein the stored information is at least used to track the web-based request during processing.

8-10. (Canceled)

11. (Currently Amended) The system of claim 1 further comprises an error-handling component that notifies a client associated with the web-based request that~~transmits a message indicating~~ processing of the web-based request ~~has been~~ halted due to a time period lapse.

12-14. (Canceled)

15. (Currently Amended) A system that employs dynamic load balancing to asynchronously process synchronous requests, comprising:

a processor that executes the following computer executable components stored on a computer readable medium:

a query management component that:

receives a web-based request from a client; and

publishes the web-based request in a queue;

an asynchronous processing component that:

detects available processing engine capacity;

predicts future processing engine capacity; and

distributes portions of the web-based request among processing engines

based on the detected and predicted processing engine capacity;

an error handling component that automatically conveys one or more portions of the web-based request associated with a failed processing engine to another processing engine, wherein the client is not informed of a processing failure;

a process[ing] engine component that groups processing engine results; and that posts synchronous requests in a message box that is accessed by one or more subscribed Web-based services that asynchronously process the synchronous requests;

an aggregating component that correlates asynchronous results with the synchronous request and groups the correlated results; and

an output component that returns the grouped processing engine results synchronous with the web-based request as a synchronous result.

16. (Currently Amended) The system of claim 15, further comprising[es] an adapter that translates the web-based request received *via* TCP/IP, IPX/SPX, UDP/IP, HTTP, SOAP, or a proprietary synchronous protocol ~~accepts a synchronous request from a client and conveys the translated web-based synchronous request to the processing engine component through an application processing interface (API).~~

17. (Previously Presented) The system of claim 16, wherein the adapter is one of a pluggable software component or an instance of an object.

18-23. (Canceled)

24. (Currently Amended) A method that facilitates Web-based asynchronous processing of synchronous requests, comprising:

employing a processor to execute computer executable instructions stored on a computer readable medium to perform the following acts:

simulating synchronous processing of a web-based request received by a client by dynamically allocating parts of the web-based request among processing engines based on present and predicted loading of the processing engines;

maintaining an association between parts of the web-based request;

grouping asynchronous results returned by the processing engines into a final result based on the maintained association between the parts of the web-based request; and

returning the final result to the client in a synchronous manner

~~accepting a synchronous request;~~

~~dynamically delineating the synchronous request across process engines based on process engine load;~~

~~correlating asynchronous results and errors; and~~

~~returning the correlated results as a synchronous result.~~

25. (Currently Amended) The method of claim 24, further comprising: [[es]]

publishing the web-based ~~synchronous~~ request in a message queue;

subscribing processing engines with the message queue; and

enabling subscribed processing engines to compete for processing portions of the web-based request.

26-35. (Canceled)

36. (New) The system of claim 7, wherein the queue is utilized to further store information indicative of at least one of a querying client, the interface component, the query management component, or the processing component.

37. (New) The system of claim 1, wherein a processing engine that is predicted to have the most processing engine capacity receives parts of the web-based request.
38. (New) The system of claim 1, wherein the processing component notifies a client associated with the web-based request of a processing engine failure.
39. (New) The system of claim 1, wherein the processing component notifies a client associated with the web-based request of failures resulting from at least one of an inoperable database, a communication channel interruption, a malicious instruction, or a system shutdown.
40. (New) The system of claim 1, wherein the processing component allocates parts of the web-based request to one or more processing engines pre-determined to receive the request based on historical actions by a client.
41. (New) The system of claim 1, wherein the processing component tracks processing of the web-based request based on connection type.
42. (New) The system of claim 17, wherein the queue is utilized to store information related to a type of connection through which the web-based request was received in order to track the web-based request during processing.
43. (New) The method of claim 24, further comprising:
continuing processing of one or more parts of a web-based request *via* failover, wherein the client is not informed of a processing failure.